Simon's Period Finding on a Quantum Annealer

Reece Robertson
Department of Computer Science & Electrical Engineering
Department of Physics
Quantum Science Institute
UMBC
Baltimore, USA
reeceroberson@umbc.edu
0000-0003-1064-0012

Emery Doucet
Department of Physics
Quantum Science Institute
UMBC
Baltimore, USA
0000-0002-2693-8553

Zakaria Mzaouali
Institut für Theoretische Physik
Universität Tübingen
Auf der Morgenstelle 14,
72076 Tübingen, Germany.
Jülich Supercomputing Centre,
Institute for Advanced Simulation
Forschungszentrum Jülich,
Wilhelm-Johnen-Straße,
Jülich, 52428, Germany.
0000-0003-3948-1318

Krzysztof Domino
Institute of Theoretical and Applied Informatics
Polish Academy of Sciences
Gliwice, Poland
0000-0001-7386-5441

Bartłomiej Gardas
Institute of Theoretical and Applied Informatics
Polish Academy of Sciences
Gliwice, Poland
0000-0002-1454-1591

Sebastian Deffner
Department of Physics
Quantum Science Institute
UMBC
Baltimore, USA
National Quantum Laboratory
College Park, USA
0000-0003-0504-6932

2025 IEEE International Conference on Quantum Computing and Engineering (QCE) | 979-8-3315-5736-2/25/$31.00 ©2025 IEEE | DOI: 10.1109/QCE65121.2025.00030

***Abstract*—Dating to 1994, Simon's period-finding algorithm is among the earliest and most fragile of quantum algorithms. The algorithm's fragility arises from the requirement that, to solve an n-qubit problem, one must fault-tolerantly sample $O(n)$ linearly independent values from a solution space. In this paper, we study an adiabatic implementation of Simon's algorithm that requires a constant number of successful samples regardless of problem size. We implement this algorithm on D-Wave hardware and solve problems with up to 298 qubits. We compare the runtime of classical algorithms to the D-Wave solution to analyze any potential advantage.**

***Index Terms*—annealing, quantum theory, error analysis**

I. Introduction

Many early quantum algorithms fall into the hidden subgroup class of algorithms, which promise exponential speedup for computational problems [1]. For instance, Deutsch's, Simon's, and Shor's algorithms solve hidden subgroup problems [2]. Shor's algorithm, in particular, took the world by storm since it solves the factoring problem with an exponential speedup over the best-known classical methods [3]. However, these algorithms strongly assume that one can access many fault-tolerant qubits. Later research has demonstrated that each hidden subgroup algorithm fails when noise is present on quantum devices [4]–[6]. Simon's algorithm, in particular, fails on gate-based hardware for problems involving over 50 qubits—or 10 qubits on hardware where swap gates are required to implement the algorithm [4]. In this paper, we analyze the performance of Simon's algorithm in the presence of noise when executed on a quantum annealer rather than a traditional quantum circuit computer. In so doing, we shift the solution frontier for this problem to approximately 300 qubits on quantum annealers (equivalent to 200 qubit problems on gate-based devices that do not require an ancillary register). This represents an improvement over the gate-based implementation; however, it is still below the performance of classical approaches. We achieve this advancement primarily by using the quantum annealer to reduce the information required to solve the problem that must be extracted from the quantum system.

Recall that Simon's algorithm relies on a two-to-one black-box function

$$f_s : \{0,1\}^n \to \{0,1\}^{n-1},$$

defined on n-bit strings, where there exists a nonzero $s \in \{0,1\}^n$ such that

$$f_s(x) = f_s(x') \iff x = x' \oplus s,$$

for all $x, x' \in \{0,1\}^n$, with $\oplus$ denoting the bitwise exclusive or (XOR) operator. Here, s is the hidden period of f, and Simon's problem consists of finding s [7]. This algorithm has practical implications in cryptography [8], [9]. Consider a system that inadvertently employs a function with the above 2-to-1 structure. In such a scenario, Simon's algorithm can efficiently determine the secret shift s, exposing a potential vulnerability. Although secure cryptographic protocols are designed to avoid such weaknesses, the conceptual insights

We thank the funding agencies listed in the acknowledgements.

979-8-3315-5736-2/25/$31.00 ©2025 IEEE
DOI 10.1109/QCE65121.2025.00030

provided by Simon's algorithm are invaluable for guiding the development of quantum-safe encryption methods.

Classically, a single query to the black-box oracle f generally tells us nothing about s. To determine s, we must repeatedly query the oracle with different input values until we find two inputs that give us the same output. Once identified, the XOR of these inputs yields s. The expected number of queries required to find two such inputs grows exponentially with the problem size n [10].

Quantum mechanically, however, in general, a single evaluation of the oracle reveals one bit of information about s. More specifically, one shot of Simon's algorithm samples a bitstring that is orthogonal to s; that is, we obtain some

$$z \in \{z \cdot s = 0 | z \in \{0,1\}^n\}, \tag{1}$$

where $\cdot$ denotes the sum of the bitwise product. As such, if we obtain $(n-1)$ linearly-independent nontrivial samples of (1), then we can construct a system of equations where the only nontrivial solution is s. In a noiseless setting, our expected number of samples grows linearly with n [10].

Of course, today's Noisy Intermediate-Scale Quantum (NISQ) computers do not constitute a noiseless setting. When noise is present in our algorithm, we are not guaranteed that our samples are drawn from (1). Moreover, if our access is limited to the black-box oracle f, we cannot distinguish between the samples that are drawn from (1) and those that are not. Thus, our classical post-processing step becomes a problem of solving a noisy system of Boolean equations. A few algorithms exist to solve such systems [11], [12]; however, the complexity of this process will likely destroy the quantum advantage for noise levels observed on a current device [13], [14].

In this work, we present an adiabatic formulation of Simon's algorithm executed on two noisy quantum annealers manufactured by D-Wave Systems. By reformulating Simon's algorithm as a Quadratic Unconstrained Binary Optimization (QUBO) problem—a representation equivalent to the Ising model used in quantum annealing [15]—we prepare a degenerate ground state that departs from encoding the traditional hidden subgroup (1). Instead, it encapsulates two specific input values z and z' from $\{0,1\}^n$ satisfying $f(z) = f(z')$ (and $z \neq z'$). This approach enables the annealing process to consistently yield both values, thereby finding the solution to Simon's problem. Additional details on the QUBO setup are provided in Section II, with performance comparisons to conventional techniques presented in Section III. An extensive overview on harnessing adiabatic quantum computation for hidden subgroup problems is available in [16]. A concrete adiabatic version of Simon's algorithm is proposed in [17]; however, its reliance on a non-Ising Hamiltonian limits its execution on quantum annealers. Moreover, several studies have extended annealing techniques to solve integer factorization [18]–[23], as well as to address challenges such as the graph isomorphism problem [24], [25] and the minimum distance problem [26].

II. Methods

The abstract formulation of Simon's algorithm on a gate-based quantum computer takes a unitary oracle U_s^f associated with the oracle function f_s and computes s. Of course, to implement Simon's algorithm on a real device, it is necessary to choose a specific oracle and to compile that oracle into a specific quantum circuit implementing U_s^f [4]. The adiabatic formulation of Simon's algorithm we propose takes a QUBO Q_s^f which encodes f_s. To this, a second oracle-independent QUBO Q^p is added which ensures that s may be reconstructed from the output samples of the annealing process.

$$\mathcal{Q}(\mathbf{x}, \mathbf{o}, \mathbf{a}|\mathbf{p}) = Q_s^f(\mathbf{x}, \mathbf{o}, \mathbf{a}) + Q^p(\mathbf{o}|\mathbf{p}). \tag{2}$$

Again, as in the gate-based case, to implement this algorithm on real hardware, it is necessary to choose a specific oracle f_s and to compile this into an explicit QUBO Q_f^s. An arbitrary boolean expression can be represented in QUBO form [27], [28], hence this is always possible.

We formulate an n-qubit input oracle for Simon's problem, where the i^{th} output bit is given by the XOR of input bits i and $(i+1)$. That is,

$$f_{\mathbf{1}}(x_1, x_2, ..., x_n)_i = x_i \oplus x_{i+1} = o_i, \tag{3}$$

where we have taken $s = \mathbf{1}$ to be the secret string of n ones. Due to the chained structure of this problem, it is representative of the simplest oracles to implement in QUBO form. Building the QUBO requires the addition of an ancilla for each XOR operation, where the value of the ancilla is given by the AND ($\wedge$) of the two input bits,

$$g_\wedge(x_1, x_2, ..., x_n)_i = x_i \wedge x_{i+1} = a_i. \tag{4}$$

As per the D-Wave documentation[1], an XOR can be encoded in the following QUBO:

$$\begin{aligned} Q_\oplus(x_1, x_2, o, a) = {} & x_1 + x_2 + o + 4a + 2x_1x_2 \\ & -2(x_1 + x_2)o - 4(x_1 + x_2)a + 4oa. \end{aligned} \tag{5}$$

We generalize this to n qubits by taking the XOR of each adjacent pair of input qubits, creating a new output and ancilla qubit with each XOR,

$$\begin{aligned} Q_{\mathbf{1}}^f(\mathbf{x}, \mathbf{o}, \mathbf{a}) = \sum_{i=1}^{n-1} & x_i + x_{i+1} + o_i + 4a_i + 2x_ix_{i+1} \\ & -2(x_i + x_{i+1})o_i - 4(x_i + x_{i+1})a_i + 4o_ia_i. \end{aligned} \tag{6}$$

In this equation, $\mathbf{x}$ is a length n vector representing the state of the input qubits, with entries x_i for $i \in [1, n]$, and $\mathbf{o}$ and $\mathbf{a}$ represent the states of the length $(n-1)$ output and ancilla registers, respectively.[2]

The spectrum of (6) for the case of $n = 3$ is shown in the first panel of Figure 1. As can be seen, all valid evaluations of the oracle (represented by the colored bars) are grouped in a degenerate ground state. In this diagram, each non-gray color represents a unique output of the oracle. Notice that these

[1] https://docs.dwavesys.com/docs/latest/handbook_reformulating.html

[2] Consult the appendix for an illustrative example of equations (6) & (7).

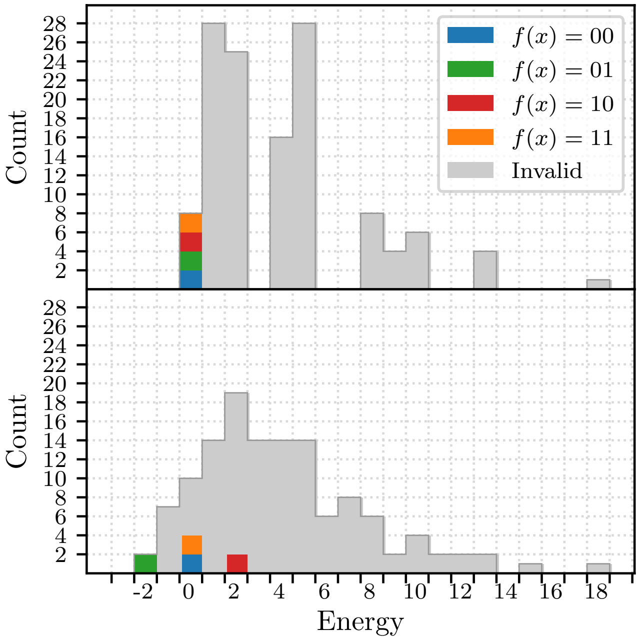

Fig. 1. Energy spectrum for oracle of size $n = 3$ (meaning there are 3 qubits in the input register, and $3n - 2 = 7$ qubits total across all registers). The top panel gives the spectrum of the QUBO with no penalties applied. In this setting, all the valid evaluations of the oracle appear in a degenerate ground state. Hence, sampling this QUBO yields one of these values at random, which queries the oracle with a random input. The bottom panel, however, gives the spectrum with penalties $p_1 = 2$ and $p_2 = -2$. In this case, two valid oracle evaluations that share a unique output are isolated in the ground state. Retrieving both inputs that map to this output is sufficient to solve the problem.

bars have two elements, precisely because the oracle is two-to-one. In other words, two valid, unique assignments of the input, output, and ancilla qubits exist for each oracle output. Finding both inputs that map to the same output (i.e., sampling both states from one of the colored bars) solves our problem. However, as we have already observed, in this configuration, all of the valid evaluations of the oracle appear in the same degenerate ground state, regardless of the value of the output qubits. Therefore, using a quantum annealer to sample from this ground state is equivalent to evaluating the oracle on a randomly chosen input (with the caveat that the input chosen for evaluation is selected miraculously after the computation of the oracle [10]). Therefore, with (6) alone, we obtain no improvement over the complexity of the classical algorithm.

However, we can do much better with a small addition to the QUBO. This modification can be made without requiring implementation knowledge of the black-box function. Consider a modification to (6) where we introduce a penalty parameter on each output qubit:

$$\begin{aligned}\mathcal{Q}(\mathbf{x},\mathbf{o},\mathbf{a}|\mathbf{p}) = \sum_{i=1}^{n-1} & x_i + x_{i+1} + (1+p_i)o_i + 4a_i + 2x_i x_{i+1} \\ & -2(x_i + x_{i+1})o_i - 4(x_i + x_{i+1})a_i + 4o_i a_i.\end{aligned} \tag{7}$$

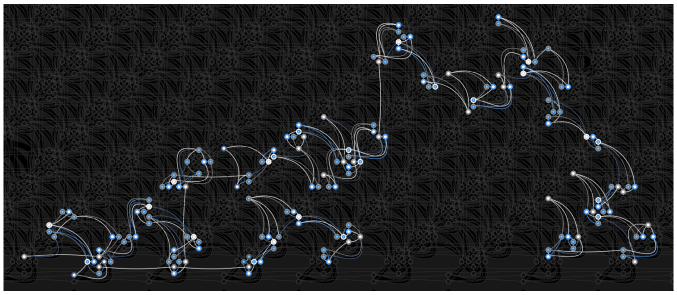

Fig. 2. The embedding of the QUBO (7) on D-Wave Advantage 5.4 for $n = 50$ (50 qubit input to the oracle, 148 total qubits). The structure of the QUBO is such that each variable interacts directly with three or six others. This connectivity pattern is sufficiently sparse to embed the problem directly on a D-Wave device without introducing swap operations or ancilla qubits.

This penalizes state transitions between output states, which effectively fixes a value of the output register throughout our computation. Significantly, state transitions between the input and ancilla registers occur with the same ease as before, with constraints applied only to enforce the rules of the oracle. The net effect of this addition is a separation of the degenerate ground state, as shown in the second panel of Figure 1 for the $n = 3$ problem with $p_1 = 2$ and $p_2 = -2$. Observe that this modification alters the degenerate ground state—it now contains the valid oracle evaluations for only a single output. Sampling both states gives us the two inputs that the oracle maps to this output, and this gives the solution to the problem.

The crux of our approach lies in applying penalty parameters to the output transitions within a QUBO formulation of Simon's problem. This allows us to solve the problem on a quantum annealer by efficiently finding two inputs that the oracle maps to the same output. We do not need to guess inputs randomly until we find two that match, nor do we need to construct a linear system of equations that we solve to find the period. We can attain the period directly by sampling both states of a precisely prepared ground state.

We evaluate these QUBOs on the D-Wave Advantage 5.4 (Pegasus) and Advantage2 Prototype 2.6 (Zephyr) quantum hardware. Each variable in the QUBO interacts with up to six other variables, allowing for a chained embedding within the hardware topology that does not require extra ancillary qubits or swap operations. See Figure 2 for an example embedding of the $n = 50$ qubit QUBO into the hardware (and recall that $n = 50$ defines the size of the oracle input, and $3n - 2 = 148$ denotes the total number of qubits utilized in the problem).

III. Results

We evaluated the performance of our adiabatic implementation of Simon's algorithm on the D-Wave Advantage and Advantage2 systems by running problem instances formulated as in (7) for various oracle sizes. Using this QUBO encoding means that the secret string was set to all 1s for all problem sizes. Systematic experiments were conducted by varying both the annealing time and the penalty parameters, with an

TABLE I
SUCCESS PERCENTAGES BY PENALTY CONFIGURATION ON ADVANTAGE2

	Balanced		Random		Uniform	
n	$p(z)$	$p(z')$	$p(z)$	$p(z')$	$p(z)$	$p(z')$
5	39.90%	58.25%	61.15%	35.68%	55.48%	41.88%
10	70.28%	18.93%	52.10%	36.40%	62.15%	24.88%
15	15.95%	45.25%	24.63%	39.68%	47.35%	6.68%
20	14.10%	38.05%	39.23%	5.73%	56.95%	1.80%
25	4.70%	30.05%	12.83%	24.30%	36.95%	3.05%
30	8.55%	8.58%	8.03%	8.05%	24.53%	3.88%
35	1.28%	20.38%	5.35%	6.18%	21.83%	0.88%
40	1.55%	7.10%	13.85%	1.03%	9.68%	0.63%
45	0.50%	9.55%	8.23%	0.23%	25.70%	0.15%
50	4.55%	1.30%	1.18%	1.28%	26.55%	0%

annealing duration of 100μs proving sufficient to solve the problem reliably over the range of sizes tested.

The proper calibration of the penalty parameters was found to be essential for the algorithm's success, with penalty values of moderate magnitude proving optimal. When the penalty magnitude was too low ($\leq$ 1) or too high ($\geq$ n), the solution space degenerated into a near-uniform sampling over all possible outputs, compromising the algorithm's discriminative ability. The performance of the algorithm is robust for intermediate penalty values; therefore, the remainder of the experiments employed a magnitude of 2.

A further observation is that ground state pairs exhibiting an imbalance in the number of 1s are unevenly sampled. Specifically, states with fewer 1s occur more frequently due to their increased resilience against local noise. To counteract this imbalance, alternating the sign of successive penalty parameters achieved a balanced distribution between the two degenerate ground states. An alternative strategy employing a random sign assignment of penalty parameters yielded comparable performance, and hence, a balanced assignment is reported here as representative of the best-case scenario.

This is illustrated in Table 1, which gives the percentage of occurrence of the two target outputs for each configuration of penalty parameters. All experiments were conducted with 4,000 shots on the Advantage2 Prototype 2.6 device. The state $p(z)$ represents the target state beginning with a 0, while the state $p(z')$ represents the target state beginning with a 1. Observe that for the uniform penalty assignment, the percentage of occurrence between both states quickly favors $p(z)$, and by $n = 50$ the state $p(z')$ is no longer observed. Under the other assignments, the percentage of occurrence between both states is more roughly comparable.

Results for penalty parameters given by $p_i = 2(-1)^{i+1}$ (the balanced configuration) are given in Figure 3. To obtain these results, we ran 4,000 shots for a total annealing time of 100μs of (7) on a D-Wave Advantage system. This plot depicts the success rate as a function of problem size, where the success rate is given by the percentage of shots that successfully sample one of the two values in the degenerate ground state. (Recall that finding both states with high probability constitutes a solution to the problem.) We find that the data is well fit by a Gaussian curve for small problem sizes ($n \leq 40$), while an exponential curve better approximates the data for larger problem sizes ($40 < n \leq 100$). Extrapolating this curve, we expect this function to scale exponentially asymptotically.

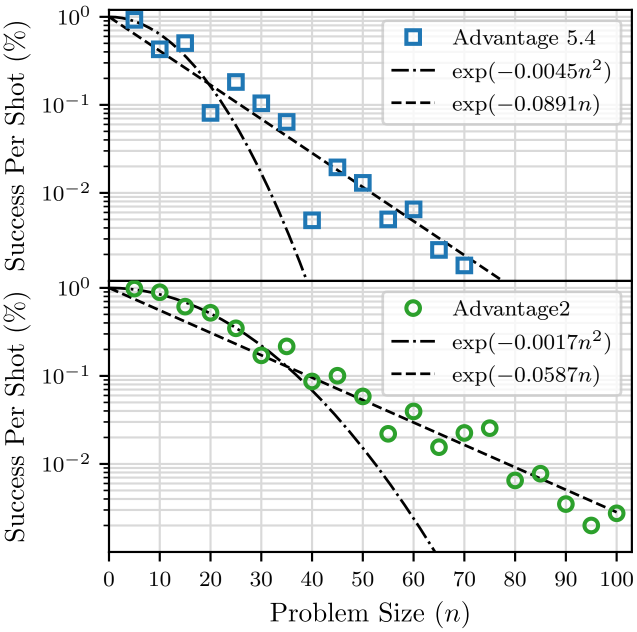

Fig. 3. Success percentage for our adiabatic Simon's algorithm as a function of problem size. As before, the problem size (n) denotes the number of qubits in the input register; the total number is $3n - 2$. The number of shots per problem is fixed at 4000, with an annealing time of 100μs. The experiment was executed on the D-Wave Advantage2 Prototype 2.6 and Advantage 5.4 systems. A successful run samples both target values in the degenerate ground state at least once. We find that for $n \leq 40$, the success rate is well fit by a Gaussian curve, whereas for $n > 40$, an exponential curve provides a better fit.

In Figure 4, we plot the approximate time to solution of our algorithm compared to a classical algorithm. To estimate the time to solution of our algorithm, we computed the expected number of shots required to sample the ground state of the Hamiltonian created by (7) at least twice. As has been discussed, with correctly selected (randomly assigned) penalty parameters, one can expect that each of the two ground states will be sampled with roughly equal probability. Therefore, in the best case, sampling the ground state twice yields a 50% chance of sampling both outputs and solving the problem. If the process fails, one can simply re-run with new random sign assignments for the penalty parameters until success.

For the classical algorithm, we utilized a QUBO solver based on tree decomposition provided by D-Wave as part of the Ocean SDK [29]. From Figure 4, we can see that the adiabatic version of Simon's algorithm scales exponentially in time, while the classical tree search solver scales quadratically. The exponential scaling of the adiabatic algorithm arises from the expected number of shots required to sample the ground state at least twice. The quadratic scaling of the tree search algorithm arises from the tree structure of the search; although the problem space doubles with every new qubit, the classical algorithm needs only to search one additional layer of the tree.

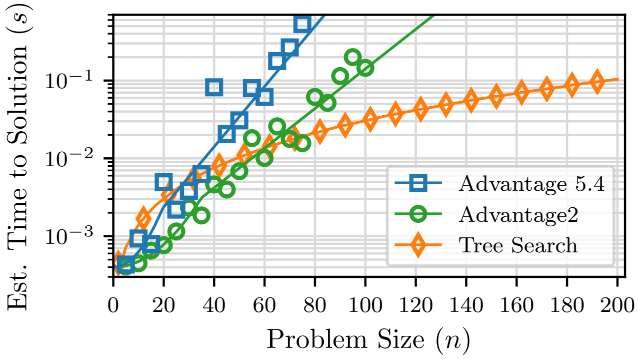

Fig. 4. Estimated time to solution as a function of problem size for our adiabatic Simon's algorithm and a classical tree search QUBO solver implemented by D-Wave. The time to solution for the annealing algorithm is estimated from the number of shots required to expect to sample from the ground state twice, yielding a 50% chance of solving the problem if the degenerate pair is equally probable. The classical algorithm manifests quadratic asymptotic scaling, while our annealing algorithm scales exponentially with problem size.

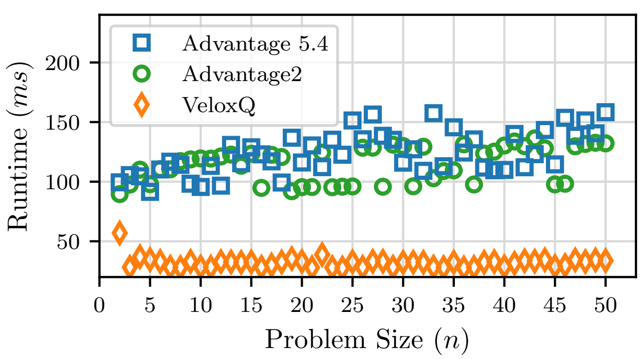

Fig. 5. Runtime as a function of problem size for our adiabatic Simon's algorithm and the VeloxQ classical QUBO solver, for $n \in [2, 50]$ and 1024 shots. As before, the experiment was executed on the D-Wave Advantage 5.4 and Advantage2 Prototype 2.6 systems. In the region tested, VeloxQ exhibited a slight downward trend in runtime, while the D-Wave systems showed upward trends. Moreover, VeloxQ consistently identified the ground state, while the rate at which D-Wave sampled from the ground state decreased with n.

This performance appears to contradict the claim that solving Simon's problem takes exponential time on a classical computer. However, one must remember two salient points. First, Simon's problem utilizes a black-box oracle, while the tree search algorithm has direct access to the instantiated oracle QUBO. Second, our chosen oracle (3) exhibits a local structure which may be exploited for a faster classical algorithm. In general, oracles need not have local structure, and the exponential runtime holds in the worst case.

In addition, the performance of the D-Wave sampling was benchmarked against VeloxQ, a novel classical solver for QUBO problems that has been shown to outperform other solvers [30]. Figure 5 compares the runtime for problem sizes $n \in [2, 50]$, using 1024 shots with a $10\mu s$ annealing time on the quantum hardware and 1000 optimization steps per shot on VeloxQ. In this range, VeloxQ achieved lower runtime—with a slight downward trend—and sampled the ground state consistently across all problem instances, whereas the D-Wave quantum annealer began to fail for $n \geq 12$. This indicates that VeloxQ outperforms even the classical tree search algorithm, obtaining a constant time to solution in the range of n tested.

This QUBO formulation of Simon's algorithm is theoretically efficient. With a perfect annealer, we can always find a sufficiently long annealing time τ such that the ground state subspace is sampled with a specified probability, say 1/2. Then, since this probability is fixed, we require $O(1)$ shots to solve Simon's problem, with the exact number of shots given as a function of the desired success probability. Therefore, the runtime scales directly with τ, which depends on the spectral properties of $\mathcal{Q}$ (e.g., for (7), the spectral gap is independent of n). On a noisy annealer, however, the presence of noise greatly hinders the runtime by reducing the probability of sampling from the ground state subspace. Future improvements in quantum hardware, alongside optimized penalty parameter strategies, may be necessary to make adiabatic implementations a competitive alternative to classical methods in solving structured computational problems.

IV. Concluding Remarks

This paper introduced a new adiabatic version of Simon's algorithm. Our algorithm prepares a degenerate ground state that encodes two inputs, which the oracle maps to the same output. Retrieving both of these inputs is sufficient to solve the problem. We implement the algorithm on a D-Wave device for problems with an input register size (n) in the range 2–100. Runtime on the device is quadratic for small problems and exponential for larger problems. Asymptotically, the runtime of our algorithm scales exponentially. We benchmarked our solution against a classical tree-based QUBO solver implemented by D-Wave and a novel classical QUBO solver called VeloxQ. The classical algorithms solved the QUBOs quickly, demonstrating a quadratic asymptotic scaling.

Our results indicate that this adiabatic implementation of Simon's algorithm solves the problem in a noisy environment with an advantage over the gate-based circuit method. In other words, this algorithm is more robust to noise than the traditional implementation. This performance advantage is realized, in part, by avoiding the expensive post-processing requirement of the gate-based method. That said, the asymptotic scaling of this adiabatic algorithm is exponential and, therefore, worse than the quadratic scaling of efficient classical algorithms for the problem. This corroborates prior results on the primacy of classical algorithms in the NISQ era [31].

This research could be continued in the future. Most notably, we implemented this algorithm for a Simon's oracle in which the hidden string is the string of all 1s. One could explore instances of Simon's problem with different hidden strings and evaluate algorithm performance. This would involve a QUBO of a more complex form than (6). While the runtime of the algorithm with a balanced assignment of penalty parameters depends upon the structure of the QUBO for the oracle of all

1s, the random assignment of penalty parameters results in an algorithm with an expected runtime that remains constant regardless of the secret string that the QUBO oracle encodes. Hence, we expect the runtime for all algorithms may increase comparably due to added complexity in the QUBO; however, it would be worthwhile to verify this conjecture.

Lastly, it should be determined how tuning the penalty term affects the scaling of the algorithm. For example, as in other combinatorial optimization problems (e.g., [13]), the spectrum of the Q may be split or overlapping depending on how the energies of various subspaces interleave for any given penalty value. Understanding how this affects the performance of the annealing process will be helpful in better understanding the efficiency of our proposed algorithm and in identifying related problems that may be amenable to similar solutions.

Acknowledgements

The authors acknowledge the Jülich Supercomputing Centre[3] for providing computing time on the D-Wave Advantage™ System JUPSI through the Jülich UNified Infrastructure for Quantum computing (JUNIQ). R.R. acknowledges funding from the UMBC Cybersecurity Institute. Z.M. acknowledges funding from the National Science Centre (NCN) under the Miniatura 8 grant: 2024/08/X/ST2/00099; and from the Ministry of Economic Affairs, Labour and Tourism Baden-Württemberg in the frame of the Competence Center Quantum Computing Baden-Württemberg (project "KQCBW24"). B.G. acknowledges funding from the National Science Centre (NCN), Poland, under Projects: Sonata Bis 10, No. 2020/38/E/ST3/00269. S.D. acknowledges support from the John Templeton Foundation under Grant No. 62422.

Appendix

To aid the understanding of equations (6) and (7), we present here the derivation of the QUBOs whose energy spectra are presented in Figure 1. These QUBOs are given by

$$\begin{aligned} &Q(x_1,x_2,x_3,o_1,o_2,a_1,a_2|p_1,p_2) = \\ &\quad x_1+2x_2+x_3+(1+p_1)o_1+(1+p_2)o_2+4a_1+4a_2 \\ &\qquad +2x_1x_2-2(x_1+x_2)o_1-4(x_1+x_2)a_1+4o_1a_1 \\ &\qquad +2x_2x_3-2(x_2+x_3)o_2-4(x_2+x_3)a_2+4o_2a_2. \end{aligned} \tag{8}$$

If the first panel of Figure 1, we have that $p_1 = p_2 = 0$, and in the second panel, we have that $p_i = 2(-1)^{i+1}$ (i.e., $p_1 = 2$ & $p_2 = -2$). It is left to the reader to verify that $Q(0,0,1,0,1,0,0|p_1,p_2)$ and $Q(1,1,0,0,1,1,0|p_1,p_2)$ are in the degenerate ground state under both penalty configurations.

References

[1] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010. [Online]. Available: https://doi.org/10.1017/CBO9780511976667

[2] R. Jozsa, "Quantum algorithms and the fourier transform," *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 454, no. 1969, pp. 323–337, 1998. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rspa.1998.0163

[3] P. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 1994, pp. 124–134. [Online]. Available: https://doi.org/10.1109/SFCS.1994.365700

[4] R. Robertson, E. Doucet, E. Spicer, and S. Deffner, "Simon's algorithm in the nisq cloud," *Entropy*, vol. 27, no. 7, 2025. [Online]. Available: https://www.mdpi.com/1099-4300/27/7/658

[5] J.-Y. Cai, "Shor's algorithm does not factor large integers in the presence of noise," *Science China Information Sciences*, vol. 67, no. 7, p. 173501, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2306.10072

[6] A. Gupta, P. Ghosh, K. Sen, and U. Sen, "Effects of noise on performance of bernstein-vazirani algorithm," 2024. [Online]. Available: https://arxiv.org/abs/2305.19745

[7] D. R. Simon, "On the power of quantum computation," *SIAM Journal on Computing*, vol. 26, no. 5, pp. 1474–1483, 1997. [Online]. Available: https://doi.org/10.1137/S0097539796298637

[8] M. Kaplan, G. Leurent, A. Leverrier, and M. Naya-Plasencia, *Breaking Symmetric Cryptosystems Using Quantum Period Finding*. Springer Berlin Heidelberg, 2016, p. 207–237. [Online]. Available: http://dx.doi.org/10.1007/978-3-662-53008-5_8

[9] T. Santoli and C. Schaffner, "Using simon's algorithm to attack symmetric-key cryptographic primitives," *Quantum Information and Computation*, vol. 17, no. 1 & 2, p. 65–78, Jan. 2017. [Online]. Available: http://dx.doi.org/10.26421/QIC17.1-2-4

[10] N. D. Mermin, *Quantum Computer Science: An Introduction*. Cambridge University Press, 2007. [Online]. Available: https://doi.org/10.1017/CBO9780511813870

[11] E. K. Alekseev, I. Oshkin, V. O. Popov, and S. V. Smyshlyaev, "Solving systems of linear boolean equations with noisy right-hand sides over the reals," *Discrete Mathematics and Applications*, vol. 28, no. 1, pp. 1–5, 2018. [Online]. Available: https://doi.org/10.1515/dma-2018-0001

[12] M. Alekhnovich, "More on average case vs approximation complexity," *computational complexity*, vol. 20, pp. 755–786, 2011. [Online]. Available: https://doi.org/10.1007/s00037-011-0029-x

[13] K. Domino, E. Doucet, R. Robertson, B. Gardas, and S. Deffner, "On the baltimore light raillink into the quantum future," 2024. [Online]. Available: https://arxiv.org/abs/2406.11268

[14] M. Koniorczyk, K. Krawiec, L. Botelho, N. Bešinović, and K. Domino, "Solving rescheduling problems in heterogeneous urban railway networks using hybrid quantum–classical approach," *Journal of Rail Transport Planning & Management*, vol. 34, p. 100521, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2210970625000186

[15] F. Glover, G. Kochenberger, R. Hennig, and Y. Du, "Quantum bridge analytics I: a tutorial on formulating and using QUBO models," *Annals of Operations Research*, vol. 314, no. 1, pp. 141–183, Jul. 2022. [Online]. Available: https://doi.org/10.1007/s10479-022-04634-2

[16] T. Albash and D. A. Lidar, "Adiabatic quantum computation," *Reviews of Modern Physics*, vol. 90, no. 1, Jan. 2018. [Online]. Available: http://dx.doi.org/10.1103/RevModPhys.90.015002

[17] I. Hen, "Period finding with adiabatic quantum computation," *EPL (Europhysics Letters)*, vol. 105, no. 5, p. 50005, Mar. 2014. [Online]. Available: http://dx.doi.org/10.1209/0295-5075/105/50005

[18] J. Suo, L. Wang, S. Yang, W. Zheng, and J. Zhang, "Quantum algorithms for typical hard problems: a perspective of cryptanalysis," *Quantum Information Processing*, vol. 19, no. 6, p. 178, Apr. 2020. [Online]. Available: https://doi.org/10.1007/s11128-020-02673-x

[19] S. Jiang, K. A. Britt, A. J. McCaskey, T. S. Humble, and S. Kais, "Quantum Annealing for Prime Factorization," *Scientific Reports*, vol. 8, no. 1, p. 17667, Dec. 2018, publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41598-018-36058-z

[20] Z. Li, N. S. Dattani, X. Chen, X. Liu, H. Wang, R. Tanburn, H. Chen, X. Peng, and J. Du, "High-fidelity adiabatic quantum computation using the intrinsic hamiltonian of a spin system: Application to the experimental factorization of 291311," 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1706.08061

[21] W. Peng, B. Wang, F. Hu, Y. Wang, X. Fang, X. Chen, and C. Wang, "Factoring larger integers with fewer qubits via quantum annealing with optimized parameters," *Science China Physics, Mechanics & Astronomy*, vol. 62, no. 6, p. 60311, Jan. 2019. [Online]. Available: https://doi.org/10.1007/s11433-018-9307-1

[22] R. Dridi and H. Alghassi, "Prime factorization using quantum annealing and computational algebraic geometry," *Scientific Reports*, vol. 7, no. 1,

[3] https://www.fz-juelich.de/ias/jsc

p. 43048, Feb. 2017, publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/srep43048

[23] N. Xu, J. Zhu, D. Lu, X. Zhou, X. Peng, and J. Du, “Quantum factorization of 143 on a dipolar-coupling nuclear magnetic resonance system,” *Phys. Rev. Lett.*, vol. 108, p. 130501, 3 2012. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.108.130501

[24] F. Gaitan and L. Clark, “Graph isomorphism and adiabatic quantum computing,” *Phys. Rev. A*, vol. 89, p. 022342, 2 2014. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevA.89.022342

[25] D. Tamascelli and L. Zanetti, “A quantum-walk-inspired adiabatic algorithm for solving graph isomorphism problems,” *Journal of Physics A: Mathematical and Theoretical*, vol. 47, no. 32, p. 325302, Jul. 2014, publisher: IOP Publishing. [Online]. Available: https://dx.doi.org/10.1088/1751-8113/47/32/325302

[26] R. Ismail, A. Kakkar, and A. Dymarsky, “A quantum annealing approach to minimum distance problem,” 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2404.17703

[27] G. Morse, T. Kozsik, O. Mencer, and P. Rakyta, “A compact qubo encoding of computational logic formulae demonstrated on cryptography constructions,” 2024. [Online]. Available: https://arxiv.org/abs/2409.07501

[28] G. Morse and T. Kozsik, “On optimal qubo encoding of boolean logic, (max-)3-sat and (max-)k-sat with integer programming,” in *Proceedings of the 7th International Conference on Algorithms, Computing and Systems*, ser. ICACS ’23. New York, NY, USA: Association for Computing Machinery, 2024, p. 145–153. [Online]. Available: https://doi.org/10.1145/3631908.3631929

[29] D-Wave, “Ocean sdk version 8.2.0 api reference: dwave-samplers,” https://docs.dwavequantum.com/en/latest/ocean/api_ref_samplers/api_ref.html#treedecompositionsolver, accessed: 4/7/2025.

[30] J. Pawłowski, J. Tuziemski, P. Tarasiuk, A. Przybysz, R. Adamski, K. Hendzel, L. Pawela, and B. Gardas, “Veloxq: A fast and efficient qubo solver,” 2025. [Online]. Available: https://arxiv.org/abs/2501.19221

[31] R. Robertson and D. Ventura, “Introducing unique: The unconventional noiseless intermediate quantum emulator,” 2024. [Online]. Available: https://arxiv.org/abs/2409.07000